Working Group 2
COTS-Based Architecture

Chair: Walt Brooks

Vice Chair: Steve Reinhardt

WG2 – Architecture: COTS-based Charter

- Charter
  - Determine the capability roadmap of anticipated COTS-based HEC system architectures through the end of the decade. Identify those critical hardware and software technology and architecture developments, required to both sustain continued growth and enhance user support.
- Chair
  - Walt Brooks, NASA Ames Research Center
- Vice-Chair
  - Steve Reinhart, SGI

WG2 – Architecture: COTS-based Guidelines and Questions

- Identify opportunities and challenges for anticipated COTS-based HEC systems architectures through the decade and determine its capability roadmap.
- Include alternative execution models, support mechanisms, local element and system structures, and system engineering factors to accelerate rate of sustained performance gain (time to solution), performance to cost, programmability, and robustness.
- Identify those critical hardware and software technology and architecture developments, required to both sustain continued growth and enhance user support.
- Example topics:
  - microprocessors, memory, wire and optical networks, packaging, cooling, power distribution, reliability, maintenance, cost, size

Working Group Participants

- Walt Brooks(chair)
- Rob Schreiber(L)
- Yuefan Deng
- Steven Gottlieb
- Charles lefurgy
- John Ziebarth
- Stephen Wheat
- Guang R. Gao
- Burton Smith

- Steve Reinhardt (co-chair)
- Bill Kramer(L)
- Don Dossa
- Dick Hildebrandt
- Greg Lindahl
- Tom McWilliams
- Curt Janseen
- Erik DeBenedicttis

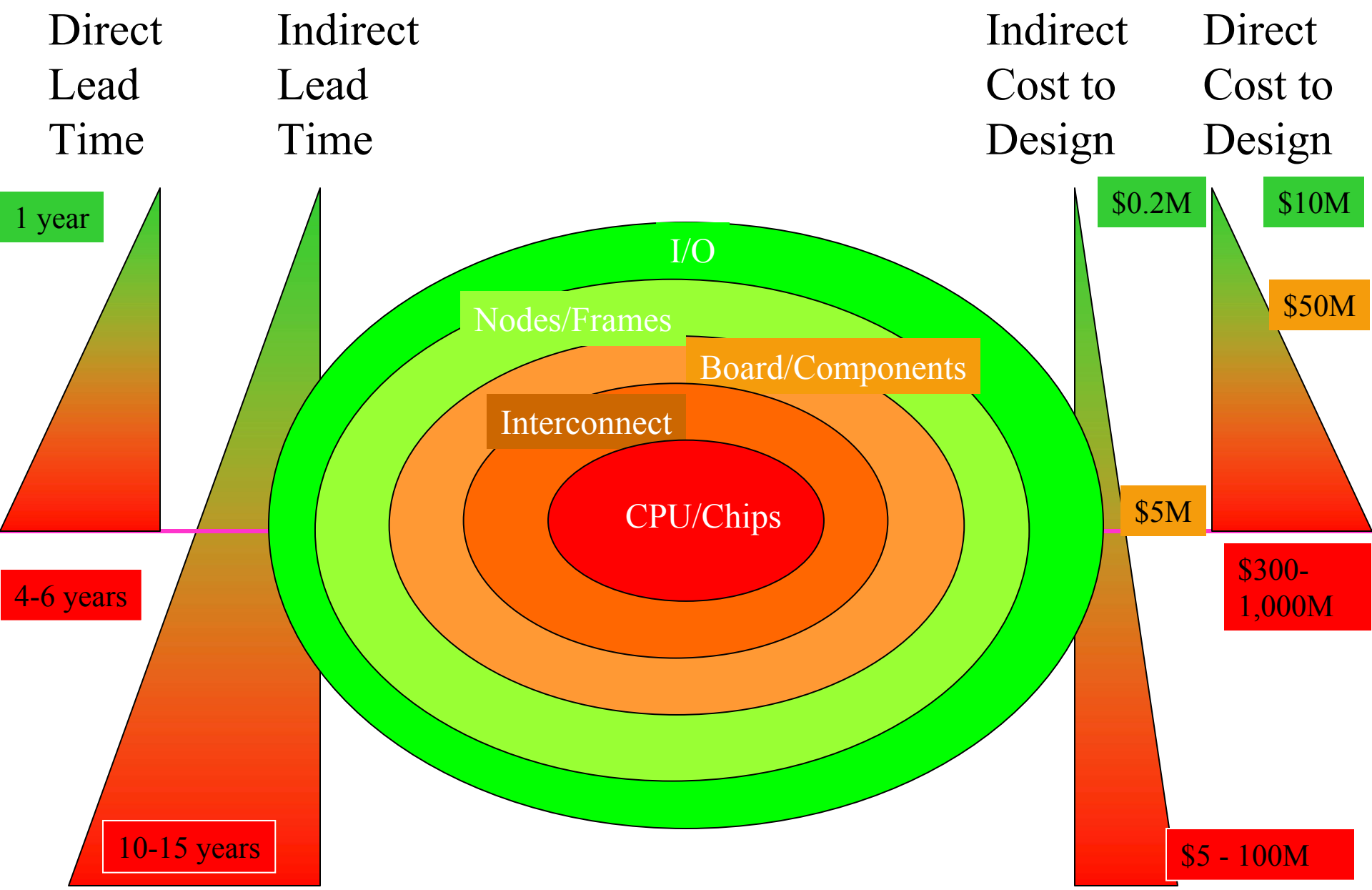
Assumptions/Definitions

- Definition of "COTS based"
  - Using systems originally intended for enterprise or individual use
  - Building Blocks-Commodity processors, commodity memory and commodity disks
  - Somebody else building the hardware and you have limited influence over
  - Examples
    - IN-Redstorm, Blue Planet, Altix
    - OUT-X1, Origins, SX-6
- Givens
  - Massive disk storage (object stores)
  - Fast wires (SERDES-driven)
  - Heterogeneous systems (processors)

Primary Technical Findings

- Improve memory bandwidth
  - We have to be patient in the short term for the next 2-3 years the die has been cast
  - Sustained Memory bandwidth is not increasing fast enough
  - Judicious investment in the COTS vendors to effect 2008
- Improve the Interconnects-"connecting to the interconnect"
  - Easier to influence than memory bandwidth
  - Connecting through I/O is too slow we need to connect to CPU at memory equivalent speeds
    - One example is HyperTransport which represents a memory grade interconnect in terms of bandwidth and is a well defined I/F -others are under development
- Provide ability for heterogeneous COTS based systems.
  - E.g. -FPGA, ASIC,… in the fabric
    - FPGA allows tightly coupled research on emerging execution models and architectural ideas without going to foundry
    - Must have the software to support programming ease for FPGA

Technology Influence



Programmatic Approaches

- Develop a Government wide coordinated method for direct Influence with the vendors to make "designs" changes
  - Less influence with COTS mfrs, more with COTS-based vendors
  - Recognize that commercial market is the primary driver for COTS
    - "Go" in early
    - Develop joint Government. research objectives-must go to vendors with a short focused list of HEC priorities
  - Where possible find common interests with the industries that drive the commodity market
  - "Software"- we may have more influence-
- Fund long Term Research
  - Academic research must have access to systems at scale in order to do relevant research
  - Strategy for moving University research into the market
- Government must be an early adopter
  - risk sharing with emerging systems

Software Issues

- Not clear that these are part of our charter but would like to be sure they are handled
  - Scaling “Linux” to 1000’s of processors
    - Administrated at full scale for capability computing
  - Scalable File systems
  - Need Compiler work to keep pace
  - Managing Open Source
    - Coordinating release implementation
    - Open source multi-vendor approach-O/S,Languages,Libraries, debuggers…
  - Overhead of MPI is going to swamp the interconnect and hamper scaling
    - Need a lower overhead approach to Message Passing

Parallel Computing

- Parallel computing is (now) the path to speed
- People think the problem is solved but it's not
- Need new benchmarks that expose true performance of COTS
- If the government is willing to invest early even at the chip level there is the potential to influence design in a way that makes scaling "commodity" systems easier
- Parallel computers to be much more general purpose than they are today
  - More useful, easier to use, and better balanced
  - Continued growth of computing may depend on it
  - To get significantly more performance, we must treat parallel computing as first class
  - COTS processors especially will be influenced only by a generally applicable approach

Themes From White Papers

- Broad Themes
  - Exploit Commodity
  - One system doesn't fit all applications-For specific family of codes Commodity can be a good solution
  - unique topology and algorithmic approaches allow exploitation of current technology
- Novel uses of current technology(Overlap with Panel 3)
  - RCM Technology- FPGA faster, lower power with multiple units-Hybrid FPGA-core is the traditional processor on chip with logic units-Need H/W architect for RCM-Apps suitable for RCM-RCM are about ease of programming
  - Streaming technology utilizing commercial chips
  - Fine grained multi threading
- Supporting Technology(Overlap with panel 1)
  - Self managing Self Aware systems
  - MRAM,EUVL,Micro-channel
  - Power Aware Computing
  - High end interconnect and scalable files systems
  - High performance interconnect technology, optical and others that can scale to large systems
  - Systems software that scales up gracefully to enormous processor count with reliability,efficiency and and ease of
  - There is a natural layering of technologies involved in a high-performance machine:
  - the basic silicon,
    - the cell boards and shared memory nodes, the cluster interconnect, the racks, the cooling, the OS kernel,
    - the added OS services, the runtime libraries, the compilers and languages, the application libraries.

Relevant White Papers

18 of the 64/80 papers have some relevance to our topic

- 6
- 10
- 12
- 16
- 17
- 31
- 33
- 39
- 45
- 46
- 47
- 50
- 65
- 68
- 72
- 75
- 80